Deep Learning Shape Priors for Object Segmentation

Fei Chen[a,c] Huimin Yu[a,b] Roland Hu[a] Xunxun Zeng[d]
[a] Department of Information Science and Electronic Engineering, Zhejiang University, China
[b] State Key Laboratory of CAD & CG, China
[c] School of Sciences, Jimei University, China
[d] College of Mathematics and Computer Science, Fuzhou University, China

Abstract

In this paper we introduce a new shape-driven approach for object segmentation. Given a training set of shapes, we first use deep Boltzmann machine to learn the hierarchical architecture of shape priors. This learned hierarchical architecture is then used to model shape variations of global and local structures in an energetic form. Finally, it is applied to data-driven variational methods to perform object extraction of corrupted data based on shape probabilistic representation. Experiments demonstrate that our model can be applied to dataset of arbitrary prior shapes, and can cope with image noise and clutter, as well as partial occlusions.

1. Introduction

Object segmentation in the presence of clutter and occlusions is a challenging task for computer vision and cross-media. Without utilizing any high-level prior information about expected objects, purely low-level information such as intensity, color and texture does not provide the desired segmentations. In numerous studies [1-6], prior knowledge about the shapes of the objects to be segmented can significantly improve the final reliability and accuracy of the segmentation result. However, given a training set of arbitrary prior shapes, there remains an open problem of how to define an appropriate prior shape model to guide object segmentation.

Early work on this problem is the Active Shape Model (ASM), which was developed by T. Cootes et al. [1]. The shape of an object is represented as a set of points. These points can represent the boundary or significant internal locations of the object. The evolutional shape is constrained by the point distribution model which is inferred from a training set of shapes. However, these methods suffer from a parameterized representation and the manual positioning of the landmarks. Later, level set based approaches have gained significant attention toward the integration of shape prior into variational segmentation [2-7]. Almost all these works optimize a linear combination of a data-driven term and a shape constraint term. Data-driven term aims at driving the segmenting curve to the object boundaries, and shape constraint term restricts possible shapes embodied by the contour.

In level set approaches, shape is represented implicitly by signed distance functions (SDF). This shape representation is consistent with the level set framework, and has its advantages since parameterization free and easy handling of topological changes. However, SDF for shape representation are not closed under linear operations, e.g., the mean shape and linear combinations of training shapes are typically no longer SDF. Most existing works only consider similar prior shapes of a known object class. For above reason, Cremers et al. [8] proposed a probabilistic representation of shape defined as a mapping $\mathbf{q}: \Omega \rightarrow [0, 1]$, that assigns to every pixel x of the shape domain Ω the probability that this pixel is inside the given shape. In particular, this relaxed definition of shape leads to convex data-driven function optimized on convex shape spaces of the following form:

$$E_i(\mathbf{q}) = \int_{\Omega} \mathbf{r}_o(x)\mathbf{q}(x)dx + \int_{\Omega} \mathbf{r}_b(x)(1-\mathbf{q}(x))dx + \int_{\Omega} \mathbf{r}_e(x)|\nabla \mathbf{q}(x)|dx \tag{1}$$

where $\mathbf{q}$ is a shape of probabilistic representation. Here $\mathbf{r}_o$ and $\mathbf{r}_b$ represent the region descriptors of the object and background, respectively, while the last term $\mathbf{r}_e$ acts as an edge indicator.

There are many ways to define the shape constraint term. Simple uniform distribution [4], Gaussian densities [2], non-parametric estimator [5, 6], manifold learning [9, 10], and sparse representation [11] were considered to model shape variation within a training set. However, most methods are recognition-based segmentation. They are suitable for segmenting objects of a known class in the image according to their possible similar shapes. If the given training set of shapes is large and associated with multiple different object classes, the statistical shape models and manifold learning do not effectively represent the shape distributions due to large variability of shapes. In addition, global transformations like translation, rotation and scaling and local transformations like bending and stretching are expensive to shape model in image

Table 1. Comparison of a number of different shape-driven models for object segmentation

Methods	Transformation		Arbitrary prior shapes
	Globally	Locally	
ASM [1]	√	√	—
Statistic models [2, 5, 6]	√	√	—
Manifold [9, 10]	√	√	—
Sparse representation [11]	√	—	√
Our method	√	√	√

segmentation. Table 1 shows the comparison of a number of different shape-driven approaches.

Recently, deep learning models [12, 13] are attractive for their well performance in modeling high-dimensional richly structured data. A deep learning model is about learning multiple levels of representation and abstraction that help to make sense of data such as images, sound, and text. The deep Boltzmann machine (DBM) has been an important development in the quest for powerful deep learning models [14, 15]. Applications that use deep learning models are numerous in computer vision and information retrieval, including classification [15], dimensionality reduction [12], visual recognition tasks [16], acoustic modeling [17], etc. Very recently, a strong probabilistic model [18] called Shape Boltzmann Machine (SBM) is proposed for the task of modeling binary object shapes. This shape generative model has the appealing property that it can both generate realistic samples and generalize to generate samples that differ from shapes in the training set.

Inspired by the above work [18], we focus on image segmentation, and propose a shape prior constraint term by deep learning to guide variational segmentation. In this paper, we first use deep Boltzmann machine to extract the hierarchical architecture of shapes in the training set. This architecture can effectively capture global and local features of prior shapes. It is then introduced into the variational framework as a shape prior term in an energetic form. By minimizing the proposed objective functional, the model is able to constrain an evolutional shape to follow global shape consistency while preserving its ability to capture local deformations.

2. Learning shape priors via DBM

A Restricted Boltzmann Machine (RBM) is a particular type of Markov Random Field (MRF) that has a two-layer architecture, in which the visible units are connected to hidden units. A Deep Boltzmann Machine (DBM) is an extension of the RBM that has multiple layers of hidden units arranged in layers [14]. In general, the shape prior can be simply described as two levels of representation: low-level local features (like edges or corners) and high-level global features (like object parts or object). Low-level local features with good invariant properties can be re-used in different object samples. On the other hand, high-level global features describe the image content, and they are more appropriate to cope with occlusion, noise, and changes on the object pose. In order to learn a model that accurately captures the global and local properties of binary shapes. We use three-layered DBM to automatically extract the hierarchical structure of shape data.

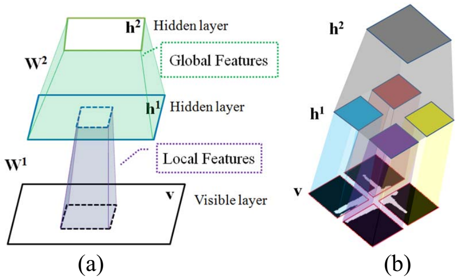

Fig. 1. Patch-based training for three-layered DBM. (a) The visible-to-hidden weights receive inputs only from a square patch of visible units below. (b) A simple case that the training shape is divided into four square patches.

Let $\mathbf{v}$ be a 2D vector of binary visible units that represent the shape, and let $\mathbf{h^1}$ and $\mathbf{h^2}$ be the lower and higher binary hidden units. The energy of the state $\{\mathbf{v}, \mathbf{h^1}, \mathbf{h^2}\}$ is defined as follows:

$$E_{DBM}(\mathbf{v}, \mathbf{h^1}, \mathbf{h^2}; \theta) \\ = -\mathbf{v^T W^1 h^1} - \mathbf{h^1}^{\mathbf{T}}\mathbf{W^2 h^2} - \mathbf{a^1}^{\mathbf{T}}\mathbf{h^1} - \mathbf{a^2}^{\mathbf{T}}\mathbf{h^2} - \mathbf{b^T v}. \quad (2)$$

Here, $\theta = \{\mathbf{W^1}, \mathbf{W^2}, \mathbf{a^1}, \mathbf{a^2}, \mathbf{b}\}$ are the model parameters. $\mathbf{W^1}$ and $\mathbf{W^2}$ represent visible-to-hidden and hidden-to-hidden symmetric interaction terms, $\mathbf{a^1}$ and $\mathbf{a^2}$ the hidden self-connections (also known as biases), and $\mathbf{b}$ is the visible self-connection. The probability that the model assigns to a visible vector $\mathbf{v}$ is

$$\mathrm{P}(\mathbf{v}; \theta) = \frac{1}{Z(\theta)} \sum_{\mathbf{h^1}, \mathbf{h^2}} \exp(-E_{DBM}(\mathbf{v}, \mathbf{h^1}, \mathbf{h^2}; \theta)). \quad (3)$$

Here, the constant $Z(\theta)$ is the partition function defined as $Z(\theta) = \sum_{\mathbf{v}} \sum_{\mathbf{h^1}, \mathbf{h^2}} \exp(-E_{DBM}(\mathbf{v}, \mathbf{h^1}, \mathbf{h^2}; \theta))$.

2.1. Learning for three-layered DBM

Given a set of aligned training shapes $\{\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_N\}$, the learning of DBM consists of determining the related weights and the biases in (2). Although exact maximum likelihood estimation of these parameters in this model is intractable, efficient approximate learning of DBMs can be carried out by using mean-field inference together with Markov Chain Monte Carlo algorithms [14]. Furthermore, the entire model can be efficiently pre-trained at each layer using RBM. Since the shapes often have similar local structural properties, the visible units can be divided into equal sized square patches to improve the learning procedure, as shown in Fig. 1a. It implied that the weights in $\mathbf{W^1}$ are restricted so that they receive inputs only from a square patch of visible units below. In order to demonstrate the advantages of three-layered DBM, we consider a simple case that the training shape is divided into four square patches for the arm posture experiment (Fig. 1b).

2.2. Approximate inference and generation

There is an efficient way of performing approximate inference of DBM. The conditional distributions over the two sets of hidden units are given by logistic functions:

$$\mathrm{P}\left(h_j^1 = 1 \middle| \mathbf{v}; \mathbf{h^2}\right) = \sigma\left(\sum_i W_{ij}^1 v_i + \sum_m W_{jm}^2 h_j^2 + a_j^1\right) \quad (4)$$

$$\mathrm{P}(h_m^2 = 1|\mathbf{h^1}) = \sigma\left(\sum_j W_{im}^2 h_i^1 + a_m^2\right) \quad (5)$$

where $\sigma(x)$ is the logistic function $1/(1+\exp(-x))$. Once binary states have been chosen for the hidden units, we can generate a shape model by setting the state of each pixel to be 1 with probability

$$\mathrm{P}(v_i = 1|\mathbf{h^1}) = \sigma\left(\sum_j W_{ij}^1 h_j^1 + b_i\right) \quad (6)$$

Although exact inference of the probability distribution $\mathrm{P}(\mathbf{v}; \theta)$ is intractable, an efficient Gibbs sampling scheme exists, as detailed in [14]. This starts with a given $\mathbf{v}$ and then alternates between updating all of the hidden features using (4) and (5), and updating all of the visible pixels using (6). After running for sufficiently long, Gibbs sampling will eventually converge to the correct solution. In practice, Gibbs sampling is doing surprisingly well with only one step. In inference procedure, bottom-up connections can be used to infer the high-level representations that would have generated an observed set of low-level features. On the other hand, top-down connections can be used to generate low-level features of shapes from high-level representations. Fig. 2 illustrates an example of approximate inference in which the generative shape model differs from training shapes. Moreover, the generative shape can be considered as global and local approximation to the training shapes.

3. Learned shape priors for segmentation

3.1. Energy formulation

In DBM, the learned weights and biases implicitly define a probability distribution over all possible binary shapes via the energy $P(\mathbf{v}, \mathbf{h^1}, \mathbf{h^2}; \theta) \propto \exp(-E_{DBM})$. Moreover, this three-layered learning can effectively capture hierarchical structures of shape priors. Lower layers detect simple local features of shape and feed into higher layers, which in turn capture more complex global features of shape. Once binary states have been chosen for the hidden units, a shape generative model can be inferred by conditional probability. Since such generative shape is defined by probability, we adopt Cremers's shape relaxed method [8] to replace the 2D visible vector $\mathbf{v}$ with a shape $\mathbf{q}$ of probabilistic representation, and define a shape constraint term in the following energetic form.

$$E_s(\mathbf{q}, \mathbf{h^1}, \mathbf{h^2}) = E_{DBM}(\mathbf{q}, \mathbf{h^1}, \mathbf{h^2}; \theta)\ . \quad (7)$$

Here, $\theta = \{\mathbf{W^1}, \mathbf{W^2}, \mathbf{a^1}, \mathbf{a^2}, \mathbf{b}\}$ are the learned parameters

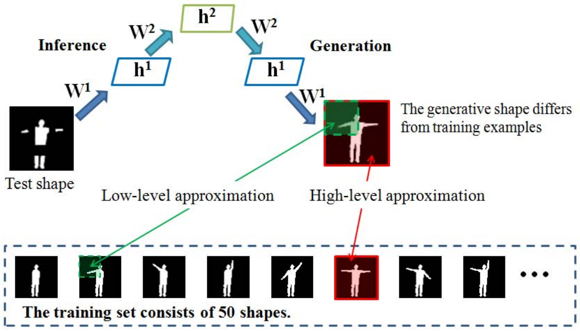

Fig. 2. Approximate inference of three-layered DBM.

of DBM, and hidden units $\mathbf{h^1}$ and $\mathbf{h^2}$ can be estimated by approximate inference. There are three obvious advantages of such shape constraint term. First, it can be applied to the dataset of arbitrary prior shapes. This three-layered learning can obtain high quality probabilistic distributions over object shapes. Second, the shape prior term encodes prior knowledge by two-layered representations of shape prior to build a flexible constraint that combines global and local structure. The numbers of hidden layers are free parameters which can be selected depending on the demands of the given task. Third, the shape constraint term is consistent with the shape probabilistic representation, and can be easily integrated into data-driven variational framework (1).

In our model, in order to relate image data feature and shape prior information, we integrate the data term with the shape constraint term towards object segmentation, by combining (1) and (7)

$$E(\mathbf{q}, \mathbf{h^1}, \mathbf{h^2}; \theta) = \underbrace{\|\nabla \mathbf{q}\|_e + \alpha_1 \mathbf{q^T r}}_{data\ term} -$$

$$\underbrace{\alpha_2 \left(\mathbf{q^T W^1 h^1} + \mathbf{h^1}^{\mathbf{T}} \mathbf{W^2 h^2} + \mathbf{a^1}^{\mathbf{T}} \mathbf{h^1} + \mathbf{a^2}^{\mathbf{T}} \mathbf{h^2} + \mathbf{q^T b}\right)}_{shape\ term} \quad (8)$$

where α_1, α_2 are positive constants, and the data term is the simplified form of (1). Here, $\|\nabla \mathbf{q}\|_e = \int_\Omega \mathbf{r}_e(x)|\nabla \mathbf{q}(x)|\mathrm{d}x$ is the weighted version of TV norm, and $\mathbf{r} = \mathbf{r}_o - \mathbf{r}_b$. To this end, we incorporate the learned shape prior into a variational framework that can account for global and local shape properties of the object to be segmented. The shape constraint term adds an additional force aiming at maximizing the similarity between the evolving shape $\mathbf{q}$ and the estimation shape inferred from the training set. Such three-layered architecture of shape term could be considered as high-level information to regularize the target shape. On the other hand, the target shape is estimated by using low-level image data. If the hidden units $\mathbf{h^1}$ and $\mathbf{h^2}$ are considered, the model (8) is generally not convex. However, the variational framework is built directly on the space of shape probabilistic representation, thus the energy functional with respect to shape $\mathbf{q}$ is convex functional over a convex set, and the monotonic convergence is guaranteed.

3.2. Energy minimization

When the learning model parameters are known, the proposed model (8) has two kinds of unknowns: the shape $\mathbf{q}$, and the DBM related hidden units $\mathbf{h^1}$ and $\mathbf{h^2}$. Instead of addressing both together, we use an alternating minimization procedure. Observe that the energy functional (8) with respect to shape $\mathbf{q}$ is convex functional over a convex set, and can be efficiently solved by Split Bregman method to obtain a global minimizer. Each layer of hidden units can be computed by mean-field approximate inference, just as done for DBM. It will be detailed in algorithm 1.

Alg. 1. Deep Learning Prior Shapes for Segmentation

```
Given the learned model parameters $\{\mathbf{W^1}, \mathbf{W^2}, \mathbf{a^1}, \mathbf{a^2}, \mathbf{b}\}$, and a new image $\mathbf{u}$ with a test shape.
Initialize $\mathbf{q}$ as mean shape of the dataset, and $\mathbf{h^2}$ as zero vector.
Repeat the following steps 1 to 3 until convergence.
1. $\mathbf{h^1} \leftarrow \sigma\left(\mathbf{q^T W^1} + \mathbf{W^2 h^2} + \mathbf{a^1}\right)$.
2. $\mathbf{q} \leftarrow \arg\min |\nabla \mathbf{q}|_e + \alpha_1 \mathbf{q^T r} - \alpha_2\left(\mathbf{q^T W^1 h^1} + \mathbf{q^T b}\right)$.
3. $\mathbf{h^2} \leftarrow \sigma\left(\mathbf{h^1}^{\mathbf{T}} \mathbf{W^2} + \mathbf{a^2}\right)$.
```

Here, $\sigma(x)$ is the element-wise logistic function $1/(1+\exp(-x))$. Due to the special structure of DBM, the cost of summing out $\mathbf{h^1}$ and $\mathbf{h^2}$ are linear in the number of hidden units. The numerical bottleneck of this segmentation algorithm is the computation of the minimizer $\mathbf{q}$ in the step 2 of the above algorithm. This can take the following form

$$\min |\nabla \mathbf{q}|_e + \alpha_1 \mathbf{q^T z}. \tag{9}$$

where $\mathbf{z} = (\mathbf{r} - \frac{\alpha_2}{\alpha_1}(\mathbf{W^1 h^1} + \mathbf{b}))$. In this work, we choose the following as the region descriptors [8].

$$\mathbf{r}_o = (c_1 - \mathbf{u})^2, \quad \mathbf{r}_b = (c_2 - \mathbf{u})^2, \quad \mathbf{r}_e = \frac{1}{1+|\nabla \mathbf{u}|} \tag{10}$$

where $c_1, c_2 \in \mathbb{R}$ represent the mean intensity inside and outside of the segmented region in image $\mathbf{u}$, respectively. Based on the probabilistic definition, it is easy to get the traditional shape region of object $\Omega_\tau = \{x: \mathbf{q}(x) \geq \tau\}$ and background of image Ω/Ω_τ by selecting a $\tau \in [0, 1]$.

We apply the Split Bregman method [20] to solve the problem (9), as was done in [21]. We first introduce the auxiliary variable, $\vec{\mathbf{d}} \leftarrow \nabla \mathbf{q}$ to add a quadratic penalty function. Then, we enforce the constraint $\vec{\mathbf{d}} = \nabla \mathbf{q}$. The resulting sequence of optimization problems is

$$\left(\mathbf{q}^{k+1}, \vec{\mathbf{d}}^{k+1}\right) = \operatorname{argmin}\left\|\vec{\mathbf{d}}\right\|_e + \alpha_1 \mathbf{q^T} \boldsymbol{z} + \frac{\lambda}{2}\left\|\vec{\mathbf{d}} - \nabla \mathbf{q} - \vec{\mathbf{e}}^k\right\|^2$$
$$\vec{\mathbf{e}}^{k+1} = \vec{\mathbf{e}}^k + \nabla \mathbf{q}^k - \vec{\mathbf{d}}^k$$

Finally, the Split Bregman approach to segmentation proceeds as follows

Alg. 2. Split Bregman method for step 2 in Alg.1.

```
Repeat the following steps 1 to 6 until $\left\|\mathbf{q}^{k+1} - \mathbf{q}^k\right\|^2 < \epsilon$.
1. Define $\mathbf{z}^k = \left(c_1^k - \mathbf{u}\right)^2 - \left(c_2^k - \mathbf{u}\right)^2 - \alpha_2(\mathbf{W^1 h^1} + \mathbf{b})$.
2. $\left(\mathbf{q}^{k+1}, \vec{\mathbf{d}}^{k+1}\right) = \operatorname{argmin}\left\|\vec{\mathbf{d}}\right\|_e + \alpha_1 \mathbf{q^T z}^k + \frac{\lambda}{2}\left\|\vec{\mathbf{d}} - \nabla \mathbf{q} - \vec{\mathbf{e}}^k\right\|^2$
3. $\vec{\mathbf{d}}^{k+1} = shrink_g(\vec{\mathbf{e}}^k + \nabla \mathbf{q}^{k+1}, \lambda)$
4. $\vec{\mathbf{e}}^{k+1} = \vec{\mathbf{e}}^k + \nabla \mathbf{q}^{k+1} - \vec{\mathbf{d}}^{k+1}$
5. Find $\Omega_\tau^k = \{x: \mathbf{q}^{k+1}(x) > \tau\}$
6. Update $c_1^{k+1} = \int_{\Omega_\tau^k} \mathbf{u}\, dx$ and $c_2^{k+1} = \int_{\Omega/\Omega_\tau^k} \mathbf{u}\, dx$
```

3.3. Transformation invariance

In applications, the target object often has similar shapes in different poses. Therefore, the shape prior should consider the transformation invariance. In DBM, the weight $\mathbf{W^1}$ consists of N columns, $\mathbf{W^1} = [W_1^1 | W_2^1 | \cdots | W_N^1]$, and each column corresponds to a 'filter' that is associated with the activation of one of the hidden units. Due to the multi-layered architecture of DBM, the weight $\mathbf{W^1}$ can be extended to incorporate transformation parameter π for transformation invariance. This can take the following form

$$\mathbf{W}_\pi^\mathbf{1} \equiv \mathbf{W^1} \circ \pi = [W_1^1 \circ \pi \mid W_2^1 \circ \pi \mid \cdots \mid W_N^1 \circ \pi] \tag{11}$$

For 2D similarity transformation, π can be parameterized as $\pi = (h_x, h_y, \rho, \sigma)$, where h_x and h_y represent the translations in x- and y-axis, ρ represents the rotation angle and σ represents the scale. Thus, the resulting optimization problem including shape alignment can be written as

$$E(\mathbf{q}, \mathbf{h^1}, \mathbf{h^2}, \pi; \theta) = \|\nabla \mathbf{q}\|_e + \alpha_1 \mathbf{q^T r} -$$
$$\alpha_2\left(\mathbf{q^T W}_\pi^\mathbf{1} \mathbf{h^1} + \mathbf{h^1}^{\mathbf{T}} \mathbf{W^2 h^2} + \mathbf{a^1}^{\mathbf{T}} \mathbf{h^1} + \mathbf{a^2}^{\mathbf{T}} \mathbf{h^2} + \mathbf{q^T b}\right) \tag{12}$$

There are three methods in shape models to deal with the parameters of π. First, the simplest, but computationally most costly method is exhaustive search. Second, gradient descent method is a general algorithm to optimize the shape energy for transformation invariance [3, 6]. However, it is difficult to balance these parameters in numerical experiments, since the optimization of the shape energy is done by local gradient descent. This optimization method will get stuck in local minima. Third, the invariance can be obtained by intrinsic alignment [5]. However, the entire set of training shapes should be normalized with respect to translation, scale and rotation in advance. In numerical experiments, it is difficult to accurately compute the center of mass and the principal axes of the shape for alignment, especially for a large number of training shapes. For some applications under an appropriate size of image region, the parameters of π are restricted to a certain domain. In our work, the exhaustive search is used because it guarantees to find the globally optimal solution.

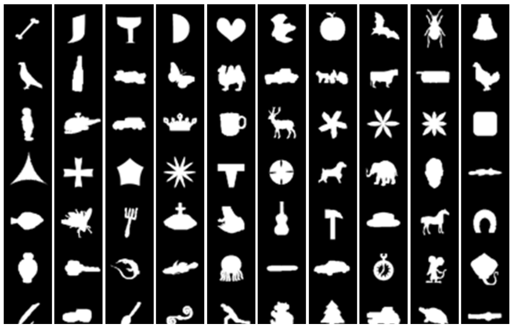

Fig. 3. Some of the shapes in MPEG-7 dataset. One image per class for the 70 classes.

4. Experimental results and validation

We present several experimental results to demonstrate the effectiveness and robustness of our model on three shape datasets: MPEG-7 dataset [22], walking-person dataset [8], and our own arm-posture dataset. Here, the Split Bregman method in Algorithm 1 is implemented in C, and called through MATLAB using a "mex" interface. All tests are done on an Intel Core i5 2.67 GHz machine.

4.1. Test on dataset MPEG-7

In this example, we create a training set of 210 shapes with 40×40 pixels selected from the part B of the MPEG-7 CE-Shape-1 dataset [22], with 3 images per object class from a total of 70 different classes. Fig. 3 shows some of the images in the dataset. The DBM trained on the dataset has 3,000 and 1,000 units in the first and second hidden layer respectively. Pre-training of DBM's requires 10,000 and 3,000 epochs for the first and second hidden layers. In addition, global training is done for 20,000 epochs. The total training time is approximately 2 days (in MATLAB).

First, in order to illustrate how the proposed model can take into account the learned shape prior for object segmentation, a synthetic image with various perturbations is used to examine the performance of our model. Several time steps in the evolution process are shown in Fig. 4. The blue contours in the evolution process are presented in the first row. The final contour matches accurately the desired shape. The second row shows the corresponding segmentation shape of probabilistic definition. The algorithm requires 30 iterations to converge, and the total computation time is about 0.76 seconds. It demonstrates two properties of our approach. First, the model is able to automatically select the reference shapes that best represent the object by approximate inference, and accurately segment the image taking into account both the image information and the shape priors. Second, our model can be applied to the training set of arbitrary shape.

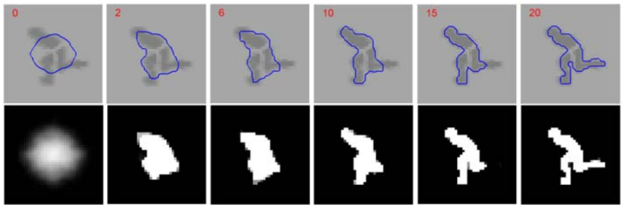

Fig. 4. The evolution of the contour (blue outlines, $\tau = 0.5$) is presented in the first row (the iteration number is shown in the upperleft corner). The second row shows the corresponding shape of probabilistic representation.

Next, various experiments are carried out to test the robustness of our model without transformation invariance, covering the case of different types of images including missing parts, noises, background clutter, etc. Here, we test our model with comparison to three recent methods: shape sparse representation [11], kernel density estimate of shape prior in variational segmentation based on shape probabilistic definition [8], kernel density estimate of shape prior for level set segmentation [6] as shown in Fig. 5. Since approximate inference of DBM can generate the shape model which differs from the training examples, the joint optimization of hidden layered parameters allows to capture the local and global shape features and obtain more satisfying results (Fig. 5b). This highlights the ability of our method not only to gather image information throughout evolution but also to accurately infer which shape is present in the image. In addition, the model with shape sparse representation is not allowed for local transformations like bending and stretching (see the bird's head in the first row of Fig. 5d) due to the target shape is approximated by sparse linear combination of training shapes. Moreover, the statistical shape models could not provide reliable shape prior information for the disturbed regions of the target shape, thus yielding dissatisfactory segmentation results (Figs. 5f and 5g). In level set approaches, shape is represented implicitly by signed distance functions. The linear combinations of shapes no longer correspond to valid shapes and then the approach in [6] could not handle the case of large shape variability of training set.

4.2. Track a walking person

In order to test our model that can account for global and local shape properties of the object to be segmented, we will apply the proposed model with shape prior to the segmentation of a partially occluded walking person. The data set we tested here is based on [8], which is publicly available. The training set (151 training shapes, walking left) is from a consecutive sequence (showing a different person walking at a different pace). Fig. 6 shows seventeen consecutive samples from the walking sequence. In order to segment a person walking in the other direction simultaneously, we construct another 151 training shapes by flipping the previous training shapes horizontally. Now, an extended training set that contains 302 shapes simultaneously encodes both walking directions. The binary shapes

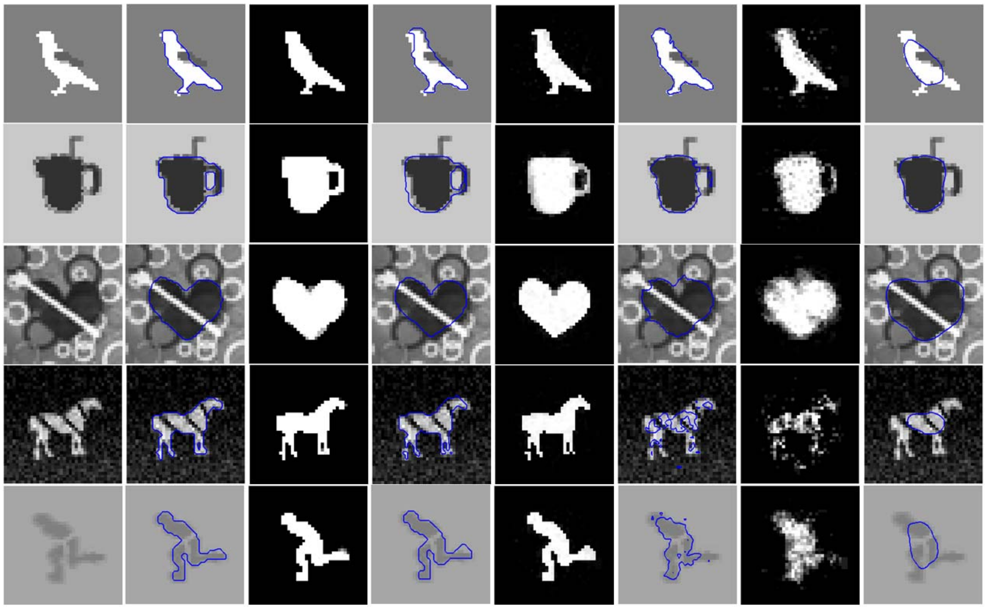

Fig. 5. Segmentation comparison of four different methods for five synthetic images with various perturbations. (a) Original images. (b) Segmentation results by the proposed model, and the corresponding shapes in (c). (d) Segmentation results by shape sparse representation [11], and the corresponding shapes in (e). (f) Segmentation results by kernel density estimate of shape prior for variational segmentation based on shape probabilistic definition [8], and the corresponding shapes in (g). (h) Segmentation results by [6].

are cropped and normalized to 50×50 pixels. The DBM trained on the dataset has 2,000 and 500 units in the first and second hidden layer respectively. Pre-training of DBM's requires 10,000 and 3,000 epochs for the first and second layers. In addition, global training is done for 30,000 epochs. The total training time is approximately 2 days.

Since the walking person here only has similar shapes in horizontal direction, we introduce artificial deformation in x direction up to 9 translation transformations in the 50×50 frame, with a step size of one pixel, i.e. $\Delta x \in \{-4,-3,-2,-1,0,1,2,3,4\}$. Fig. 7 provides segmentation comparisons obtained on a sequence showing a person walking in different directions. The initial translation parameter in the first frame is given manually. When moving on to a new frame, the final translation parameter of previous frame is used as the initial translation estimate for the segmentation of the current frame. The experiments show that our model takes about 4 seconds to process one frame. By visual inspection, the nearly similar results are obtained by using the proposed model with deep learned shape prior (Fig. 7b) in contrast to the model with shape sparse representation (Fig. 7d). Because the training shapes are binary and the multi-layered learning can effectively capture global structures of shape priors, the resulting shape of the proposed model is close to binary shape (Fig. 7c) and more accurate at the edges in contrast to other methods. Moreover, our results compare favorably to those obtained by existing statistical methods (Figs. 7f and 7h). Encoding multiple different classes of shape samples, the statistical distributions could not provide reliable shape prior information for segmentation, e.g., they could not provide a correct walking direction in Fig. 7g.

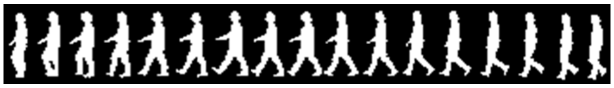

Fig. 6. Seventeen consecutive samples from the walking sequence.

4.3. Arm posture segmentation

As a third example, we consider arm-posture images, with varying positions of the arms. Our own training dataset for this example consists of 50 binary shapes with 60×60 pixels, as shown in Fig. 8. The DBM trained on the dataset has 200×4 and 200 units in the first and second hidden layer respectively. Pre-training of DBM's requires 5,000 epochs for the two hidden layers. Global training is done for 30,000 epochs. The total training time is approximately one day. For different people, the shape of arm posture often has certain variations, or the target object has similar shapes in different poses. To demonstrate the robustness of our approach for real images, several experiments about arm posture segmentation are carried out. As seen in Fig. 9, variational model with deep learned shape prior makes the segmentation process robust to missing and misleading information (Figs. 9b). Such multi-layered shape prior can combine bottom-up and top-down processing of an image to model shape variations of global and local structures. Due to a small sample size and large variation of samples,

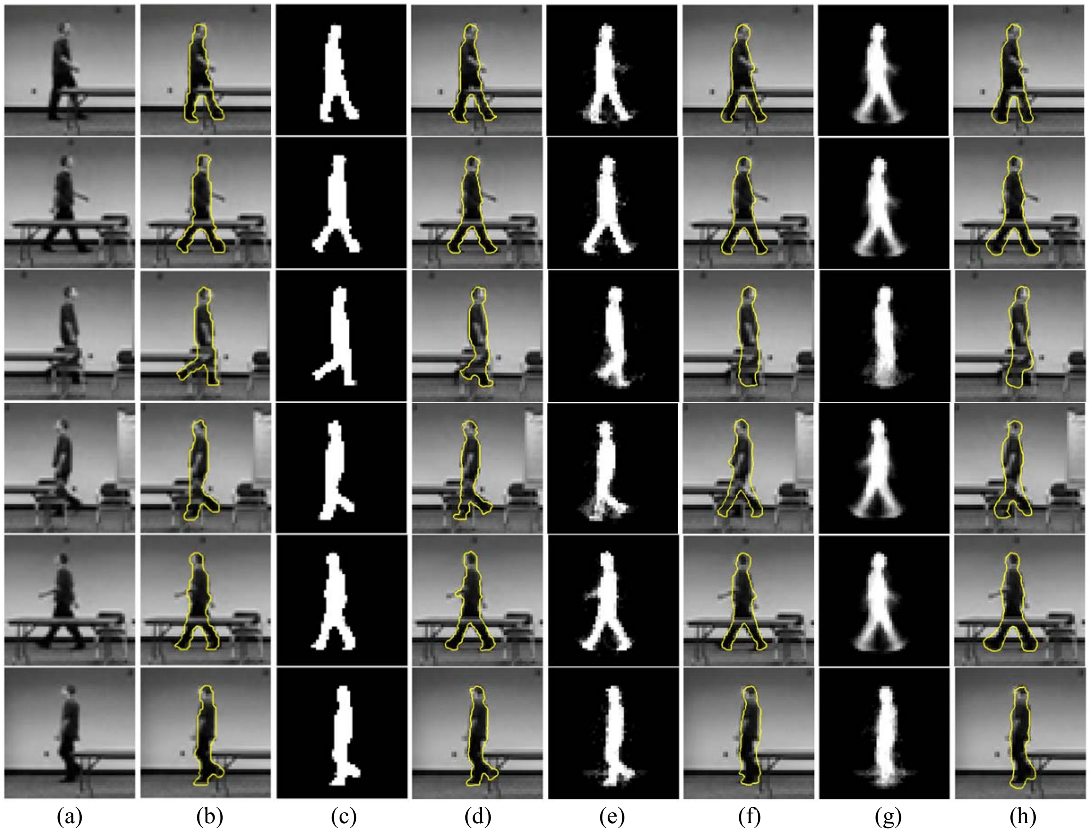

Fig. 7. Typical comparison of six frames based on the enlarged training set of 302 shapes. (a) Original images. (b) Segmentation results by the proposed model, and the corresponding shapes in (c). (d) Segmentation results by shape sparse representation [11], and the corresponding shapes in (e). (f) Segmentation results by kernel density estimate of shape prior for variational segmentation based on shape probabilistic definition [8], and the corresponding shapes in (g). (h) Segmentation results by [6].

shape sparse representation (Figs. 9d) and the statistic shape models (Figs. 9f and 9h) model cannot effectively capture the local edge variations, and yield dissatisfactory segmentation results.

5. Conclusion

Our approach consists of two stages. The first is the use of deep Boltzmann machine to extract the hierarchical structure of the training shapes. This hierarchical structure can effectively capture global and local structures of prior shapes. During the second stage a shape-driven variational framework is built directly on the space of shape probabilistic representation. This hierarchical structure of shape prior is introduced in an energetic form to regularize the target shape in variational image segmentation. We demonstrate the effectiveness of the resulting algorithm in segmenting images that involve low-quality data and occlusions.

Acknowledgment

This work was supported by a National Key Basic Research Project of China (973 Program No. 2012CB316400).

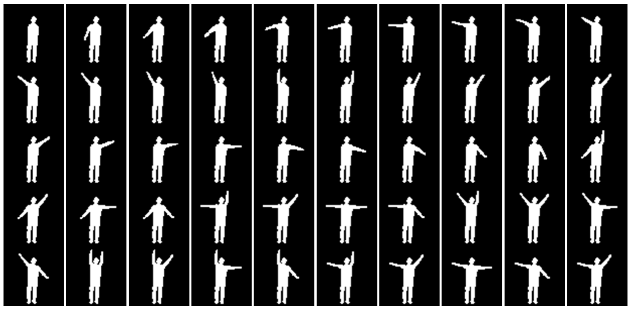

Fig. 8. An arm-posture dataset of 50 shapes.

References

[1] T. Cootes, C. Taylor, D. Cooper and J. Graham, Active shape models—their training and application, Comput. Vision Image Understanding, 61 (1):38–59, 1995.

[2] M. Leventon, W. Grimson, and O. Faugeras, Statistical shape influence in geodesic active contours, in CVPR, 2000.

[3] M. Rousson and N. Paragios, Shape priors for level set representations, in ECCV, 2002.

[4] A. Tsai, et al., A shape-based approach to the segmentation of medical imagery using level sets, IEEE Trans. medical imaging, 22(2):137-154, 2003.

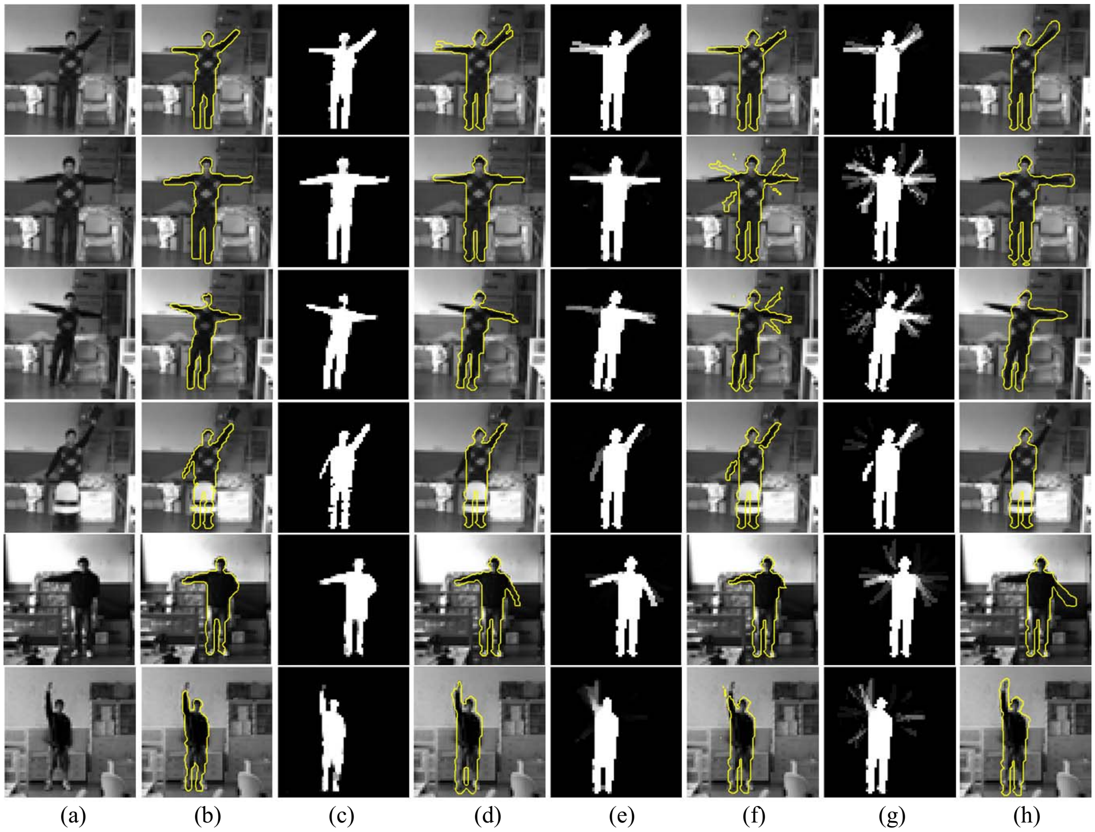

Fig. 9. Typical segmentation comparisons of four different methods. (a) Original images. (b) Results by our model, and the corresponding shapes in (c). (d) Results by shape sparse representation [11], and the corresponding shapes in (e). (f) Results by kernel density estimate of shape prior based on shape probabilistic definition [8], and the corresponding shapes in (g). (h) Segmentation results by [6].

[5] D. Cremers, S. Osher, and S. Soatto, Kernel density estimation and intrinsic alignment for shape priors in level set segmentation, Int'l J. Computer Vision, 69:335-351, 2006.

[6] M. Rousson and D. Cremers, Efficient kernel density estimation of shape and intensity priors for level set segmentation, MICCAI, 3750:757-764, 2005.

[7] K. Fundana, N. Overgaard, and A. Heyden, Variational segmentation of image sequences using region-based active contours and deformable shape priors, Int'l J. Computer Vision, 80:289-299, 2008.

[8] D. Cremers, F. Schmidt, and F. Barthel, Shape priors in variational image segmentation: convexity, lipschitz continuity and globally optimal solutions, in CVPR, 2008.

[9] Y. Rathi, N. Vaswani, and A. Tannenbaum, A generic framework for tracking using particle filter with dynamic shape prior, IEEE Trans. image processing, 16(5):1370-1382, 2007.

[10] V. Prisacariu and I. Reid, Nonlinear shape manifolds as shape priors in level set segmentation and tracking, in CVPR, 2011.

[11] F. Chen, H. Yu and R. Hu, Shape sparse representation for joint object classification and segmentation, IEEE Trans. image processing, 22(3):992-1004, 2013.

[12] G. Hinton and R. Salakhutdinov, Reducing the dimensionality of data with neural networks, Science, 313(28):504-507, 2006.

[13] G. Hinton, S. Osindero and Y. Teh, A fast learning algorithm for deep belief nets, Neural Computation, 18:1527-1554, 2006.

[14] R. Salakhutdinov and G. E. Hinton, Deep Boltzmann machines, in 12th International Conference on AI and Statistics, 2009.

[15] R. Salakhutdinov and G. Hinton, An efficient learning procedure for deep Boltzmann machines, in 13th International Conference on AI and Statistics, 2010.

[16] H. Lee , R. Grosse , R. Ranganath, and A. Y. Ng, Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations, ICML, 2009.

[17] A. Mohamed G. Dahl, M. Ranzato and G. Hinton, Phone recognition with the mean-covariance restricted boltzmann machine, in NIPS, 2010,

[18] S. Ali Eslami, N. Heess, and J. Winn, The shape Boltzmann machine: a strong model of object shape, in CVPR, 2012.

[19] T. Chan and L. Vese, Active contours without edges, IEEE Trans. image processing, 10:266-277, 2001.

[20] T. Goldstein and S. Osher, The split bregman method for l1-regularized problems, SIAM J. Image Sciences, 2(2):323–343, 2009.

[21] T. Goldstein, X. Bresson, and S. Osher, Geometric applications of the split Bregman method: segmentation and surface reconstruction, J Sci Comput,45: 272–293, 2010.

[22] L. Latecki, R. Lakamper, and U. Eckhardt, Shape descriptors for non-rigid shapes with a single closed contour, in CVPR, 2000.